Crawling The Web for Libre: Selecting, Integrating, Extending and Releasing Open Source Software

Ian Truslove, Ruth E. Duerr, Hannah Wilcox, Matthew H Savoie, Luis Lopez, Michael Brandt

http://nsidc.org/libre

Introduction

The National Snow and Ice Data Center (NSIDC) supports research into our world's frozen realms: the snow, ice, glaciers, frozen ground, and climate interactions that make up Earth's cryosphere.

Libre is a project developed by NSIDC, devoted to liberating science data from its traditional constraints of publication, location, and findability. Libre embraces and builds on the notion of making knowledge freely available, and both Creative Commons licensed content and Open Source Software are crucial building blocks for, as well as required deliverable outcomes of the project.

One important aspect of the Libre project is to discover cryospheric data published on the internet without prior knowledge of the location or even existence of that data. Inspired by well-known search engines and their underlying web crawling technologies, Libre has explored tools and technologies required to build a search engine tailored to allow users to easily discover geospatial data related to the polar regions.

This poster recounts the Libre team's experiences selecting, using, and extending Apache Nutch, a popular Open Source Software (OSS) web search project.

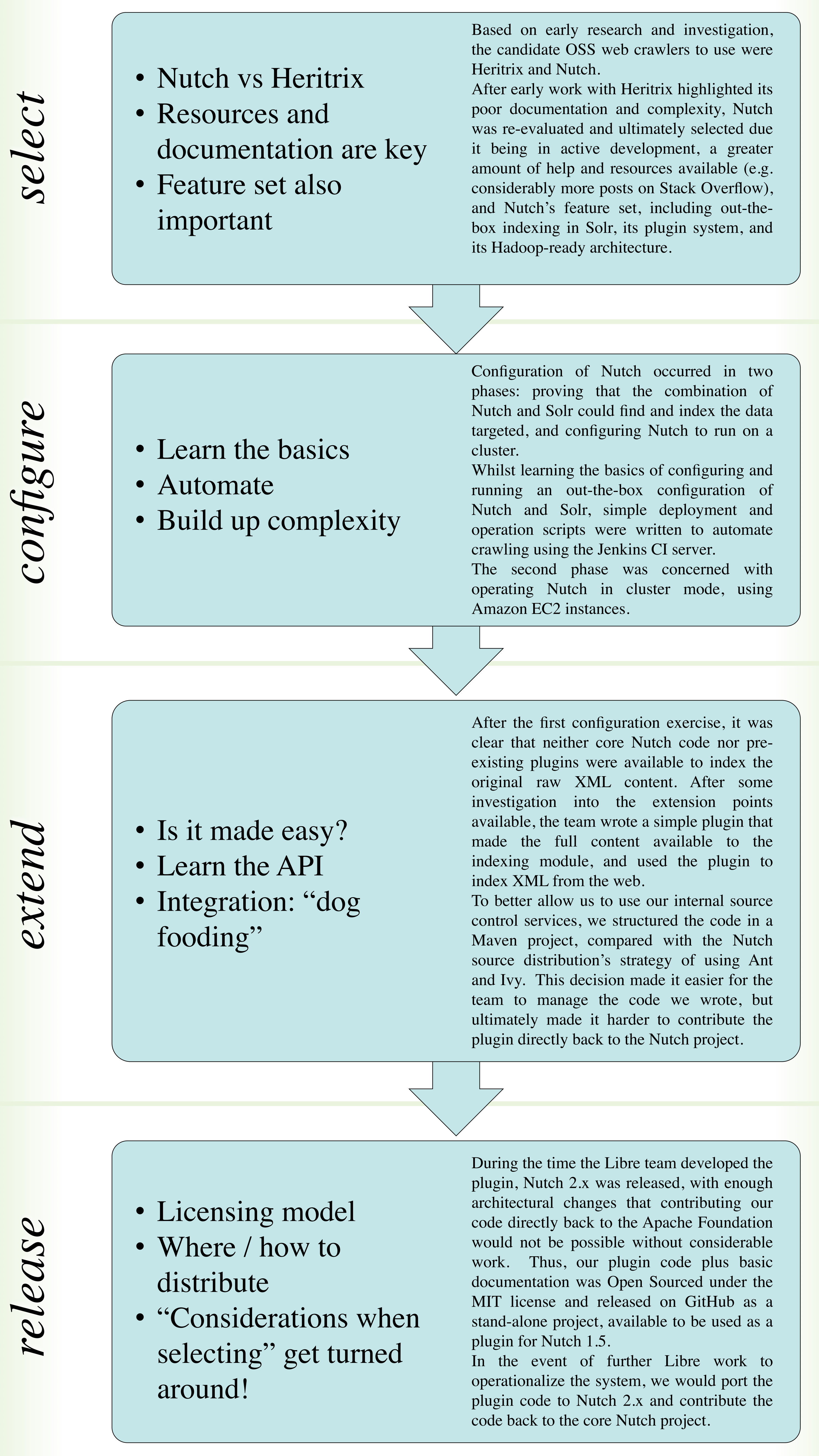
Considerations when selecting an Open Source Software project

Usage	• Do you plan on simply using, or extending the software?
Options	• There are likely a number of OSS solutions; how do they compare?
Implementation Language	• If you plan on extending, is the language familiar? Is the codebase well-documented and well-tested?
Architecture	• Is there a plugin architecture for extension?
Maturity	• Newer projects may still have more bugs • Older projects may lack newer features, and may suffer code rot.
Vitality	• Is there still an active developer community working on the project? Look for recent updates in wikis, bug trackers, mailing lists, or in the source control.
Size	• How large is the project? How many developers? • Sites such as Ohloh provide statistics.
Documentation	• What is the state of the user and developer documentation? How critical is this to your adoption?
Tools	• Does the project have a supporting infrastructure, e.g. a bug tracker, mailing list, wiki, unit tests, all with recent activity?

Links and Resources

- NSIDC: http://nsidc.org
- Libre: http://nsidc.org/libre
- Nutch: http://nutch.apache.org
- Heritrix: https://webarchive.jira.com/wiki/display/Heritrix
- Ohloh: http://www.ohloh.net
- Libre Raw XML plugin: https://github.com/nsidc/libre-nutch-raw-xml-plugin
- This poster: http://goo.gl/yLp2U

Developing a "Google for Data"



select

- Nutch vs Heritrix
- Resources and documentation are key
- Feature set also important

Based on early research and investigation, the candidate OSS web crawlers to use were Heritrix and Nutch.

After early work with Heritrix highlighted its poor documentation and complexity, Nutch was re-evaluated and ultimately selected due it being in active development, a greater amount of help and resources available (e.g. considerably more posts on Stack Overflow), and Nutch's feature set, including out-the-box indexing in Solr, its plugin system, and its Hadoop-ready architecture.

configure

- Learn the basics
- Automate
- Build up complexity

Configuration of Nutch occurred in two phases: proving that the combination of Nutch and Solr could find and index the data targeted, and configuring Nutch to run on a cluster.

Whilst learning the basics of configuring and running an out-the-box configuration of Nutch and Solr, simple deployment and operation scripts were written to automate crawling using the Jenkins CI server.

The second phase was concerned with operating Nutch in cluster mode, using Amazon EC2 instances.

extend

- Is it made easy?
- Learn the API
- Integration: "dog fooding"

After the first configuration exercise, it was clear that neither core Nutch code nor pre-existing plugins were available to index the original raw XML content. After some investigation into the extension points available, the team wrote a simple plugin that made the full content available to the indexing module, and used the plugin to index XML from the web.

To better allow us to use our internal source control services, we structured the code in a Maven project, compared with the Nutch source distribution's strategy of using Ant and Ivy. This decision made it easier for the team to manage the code we wrote, but ultimately made it harder to contribute the plugin directly back to the Nutch project.

release

- Licensing model
- Where / how to distribute
- "Considerations when selecting" get turned around!

During the time the Libre team developed the plugin, Nutch 2.x was released, with enough architectural changes that contributing our code directly back to the Apache Foundation would not be possible without considerable work. Thus, our plugin code plus basic documentation was Open Sourced under the MIT license and released on GitHub as a stand-alone project, available to be used as a plugin for Nutch 1.5.

In the event of further Libre work to operationalize the system, we would port the plugin code to Nutch 2.x and contribute the code back to the core Nutch project.

Libre Crawler goals

The Libre Crawler is intended to be a system capable of discovering the majority of cryospheric data published on the Internet. In particular, the Crawler should find the following:

- OpenSearch Description Documents
- OGC "getCapabilities" documents
- OAI-PMH metadata feeds
- ESIP Collection and Data Cast feeds

Rough estimates based on the number of US educational and government domains indicated a crawl frontier size of ~1 million websites, and ~100 million pages.

The first phase of this work was to prototype an architecture capable of crawling this portion of the web on a monthly basis, and finding and indexing any "interesting" data.

Crawling the Web with Nutch and Amazon Web Services

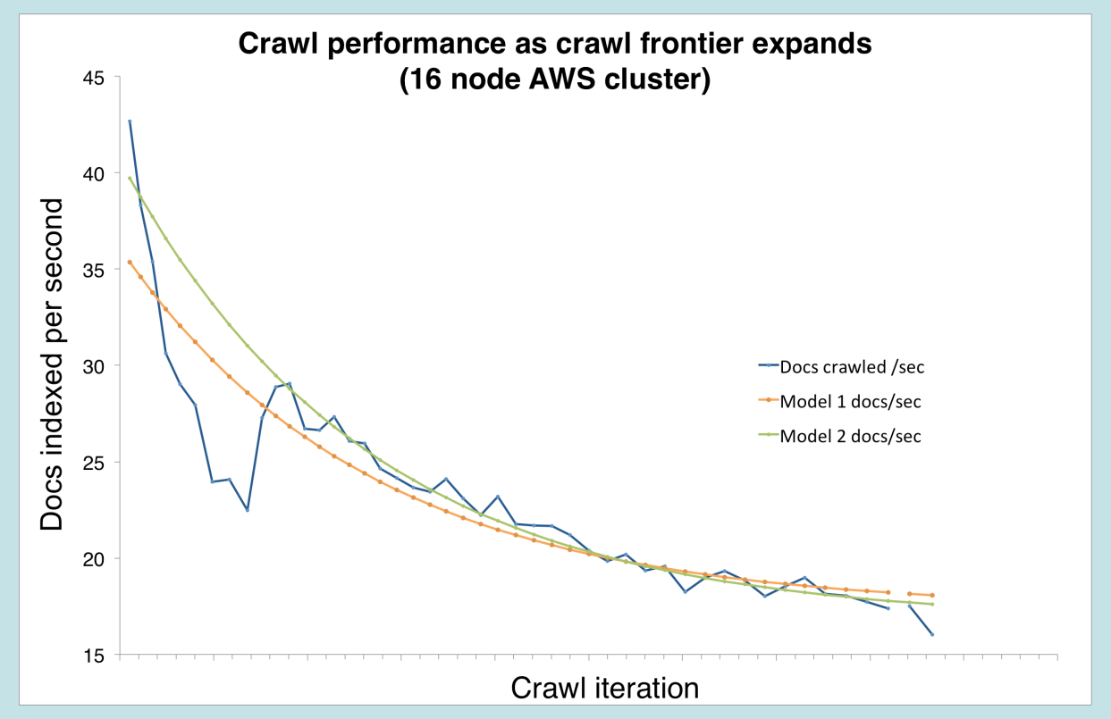
architecture

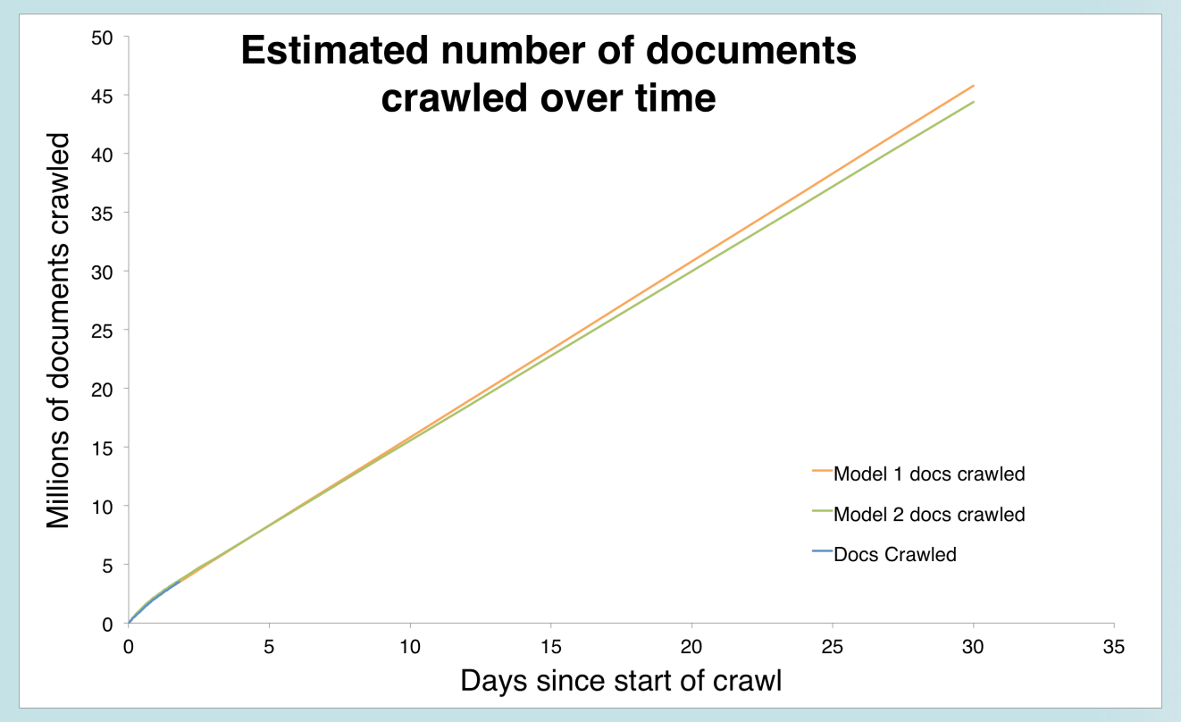
Performance of the first Nutch experiments clearly indicated the need to scale the crawling architecture to meet the estimated performance goals of indexing ~100 million pages per month.

To this end, the Libre team configured a Hadoop cluster using Amazon's Elastic Compute Cloud (EC2), with one job tracker, one Solr instance, and between four and sixteen worker nodes.

Nutch is built on the Apache Hadoop framework, and is well suited to scaling with large numbers of machines.

performance





All of the cluster sizes tested showed a clear performance degradation as the number of documents crawled increased.

Modeling the crawl performance curve of the 16-node cluster using a decay function, the extrapolated curve indicates a potential 50 million documents indexed in a one month period.

The curve was modeled with:

$$y = a\exp(-bt) + c$$

Model 1:

- a=18.73031232
- b=0.074999128
- c=17.34365199

Model 2:

- a=23.94988278
- b=0.075555209
- c=16.70013544

next steps

Steps required to develop the prototype into a fully operational web crawler include:

- Re-implementing the Raw XML indexer in Nutch 2.x
- Further investigation into performance characteristics at scale (and optimizations therein), particularly of the LinkDB
- Development of crawl frontier management strategies and algorithms
- Development of a query interface, providing access to the data discovered by the crawler

Supporting Awards: NASA NNX10AB07A
NSF ARC 0946625